

MẠCH THANH GHI (SHIFT REGISTER)

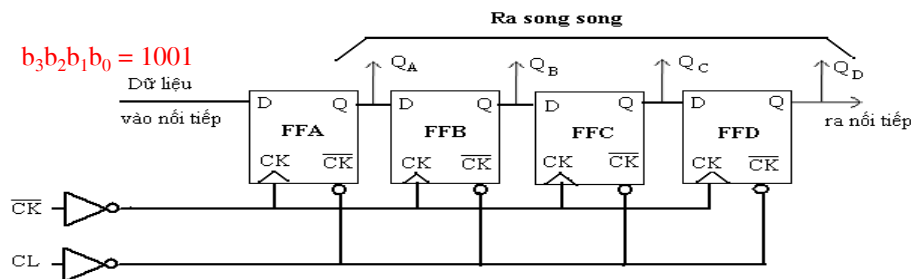
I. Đại cương



- Mỗi Flip Flop có 2 trạng thái 0 hay 1 và ta có thể kích thích vào một trong hai trạng thái đó như ý muốn. Các ngõ ra chỉ thay đổi khi ta bắt buộc thay đổi. Ta nói FF có đặc tính ký ức.
- Nếu ta dùng nhiều FF ta có thể ghi vào chuỗi số nhị phân n bit với n là số FF và bit là đơn vị của FF.
- Nhóm FF dùng vào công việc này để thành lập mạch đăng ký. Nhóm FF dùng vào việc điều khiển này thành lập mạch đăng ký di chuyển (mạch thanh ghi).

II. Dữ liệu vào theo lối nối tiếp

Ta dùng 4 FF dưới dạng FF-D được mắc như ở hình vẽ.



- Trước tiên áp xung vào **CL** để xóa các **FF**.
- Tạo xung nạp bit b_0 vào ghi dịch $Q_A=1, Q_B=Q_C=Q_D=0$.
- Cạnh lên xung tiếp theo, bit b_1 vào Q_A , bit b_0 ở Q_A tức $Q_A = D_B$ nên sau xung hai $Q_A=b_1=0, Q_B=b_0=1, Q_C=Q_D=0$.
- Như vậy sau ba xung $Q_A=b_2=0, Q_B=b_1=0, Q_C=b_0=1, Q_D=b_0=0$, sau 4 xung

$$Q_A=b_3=1, Q_B=b_2=0, Q_C=b_1=0, Q_D=b_0=1$$

Vậy sau 4 xung 4 bit là 0011 đã truyền vào ghi dịch. Và ta có 4 ngõ ra song song. Và các tín hiệu này sẽ lưu trữ vĩnh viễn nếu ngưng đồng hồ dữ liệu cho tới khi ta tắt máy

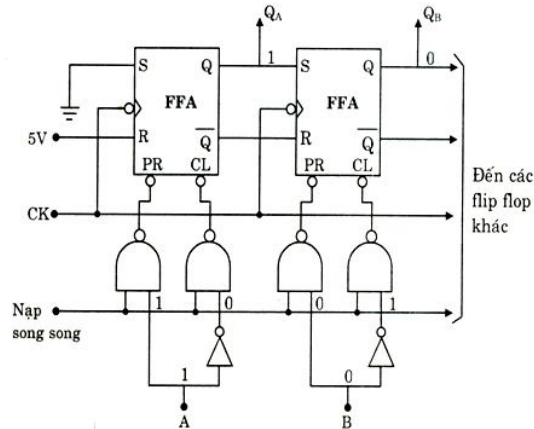
CLEAR	FF0	FF1	FF2	FF3
	0	0	0	0

- Ban đầu các ngõ ra 4 FF bằng 0 (nối clear xuống mass), ta tuần tự cho dữ liệu ở ngõ vào nối tiếp. Khi có một cạnh lên của xung CK dữ liệu đi vào trong mạch đăng ký di chuyển 1 bit, với dữ liệu n bit thì sau n xung CK dữ liệu nằm hoàn toàn trong mạch đăng ký di chuyển.
- Ngõ ra song song: ngõ ra Q của các FF.
- Muốn lấy ra theo lối nối tiếp ta phải tiếp tục cho xung CK vào và sau n xung CK thì dữ liệu hoàn toàn ra khỏi mạch đăng ký di chuyển.

DL vào	CK	Q_A	Q_B	Q_C	Q_D
1	↑	0	0	0	0
0	↑	1	0	0	0
1	↑	0	1	0	0
1	↑	1	0	1	0
1	↑	1	1	0	1
1	↑	1	1	1	0
1	↑	1	1	1	1
1	↑	1	1	1	1
1	↑	1	1	1	1

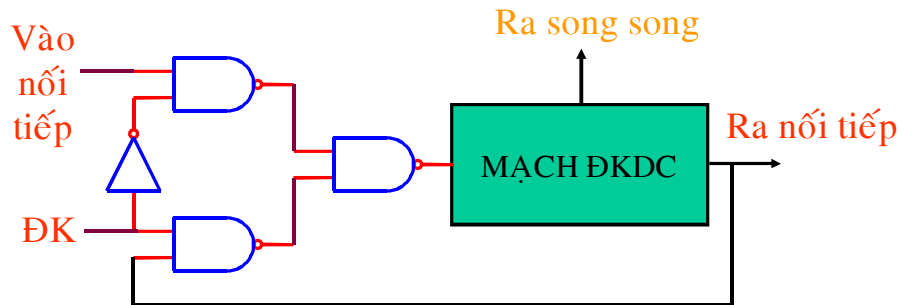
III. Dữ liệu vào theo lối song song

- Ngõ nạp song song ở **thấp** : dữ liệu ở A và B không vào các FF.
- Ngõ nạp song song lên **cao** : dữ liệu ở A và B được áp vào các ngõ vào PR và CL : FF thay đổi trạng thái theo A và B. CK không có tác dụng.
- Cách thức trên cho phép nạp vào bit 0 hay 1 tùy ý.



Muốn cho dữ liệu vào theo lối song song, ta phải đặt sẵn dữ liệu tại các ngõ A, B. **Khi nạp song song lên 1 ta thấy:**

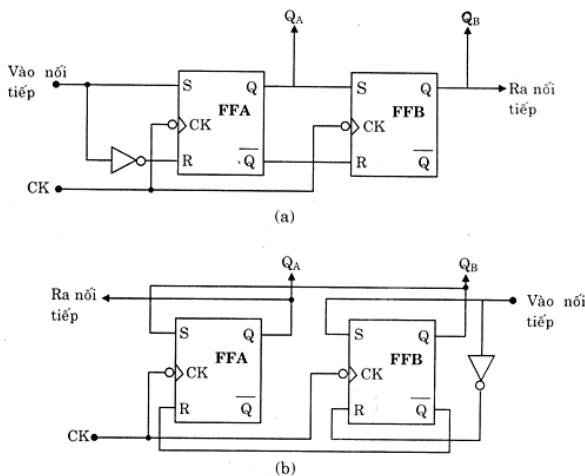
- Nếu $A = 1$ nên $Q_A = 1$.
 - Nếu $B = 0$ nên $Q_B = 0$. Ta nói dữ liệu từ ngõ vào được đưa đến ngõ ra khi ngõ điều khiển nhận vào lên 1.
 - Ngõ đồng hồ CK không có tác dụng khi nạp song song.
- o Muốn ngõ ra theo lối nối tiếp, ta lấy ở ngõ cuối cùng.
 - o Muốn mạch dịch trái, ta phải nối các đường hồi tiếp Q_B về ngõ vào A và ngõ ra Q_A chính là ngõ ra nối tiếp, ngõ vào nối tiếp chính là ngõ vào B.
 - o Khi lấy ra theo nối tiếp thì dữ liệu sẽ bị mất đi. Muốn dữ liệu không bị mất, ta nối đường hồi tiếp từ ngõ ra cuối cùng trở về ngõ vào đầu.
 - o Ta cần thêm 1 số cổng logic để thiết kế dữ liệu vào theo lối nối tiếp, thiết kế dữ liệu vào và ra theo lối song song, hoặc dữ liệu ra nối tiếp và nối đường hồi tiếp để mạch chạy tuần hoàn.



$\text{ĐK} = 0$: vào nối tiếp

$\text{ĐK} = 1$: hồi tiếp (chạy tuần hoàn), 74164

Mạch dịch phải & dịch trái:



Hình 6.45: Dịch chuyển dữ liệu đã nạp: (a) dịch chuyển phải; (b) dịch chuyển trái

Ta mắc thêm các cổng logic để tạo dịch trái hay phải mà chỉ cần thay đổi cách mắc dây nối bên ngoài một cách đơn giản hay dùng bit điều khiển thích hợp.

Thí dụ:

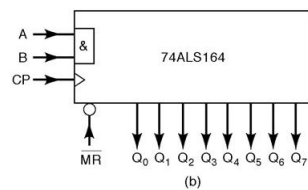
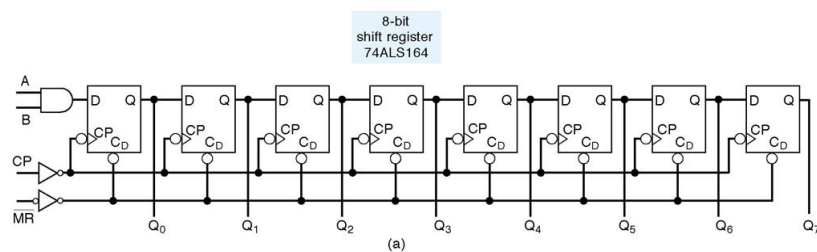
7494 : 4 bit, vào // hay nối tiếp – ra nối tiếp

7495 : 4 bit, vào // hay nối tiếp – ra // hay dịch phải trái.

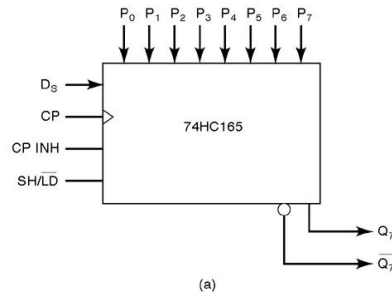
74164: 8 bit, vào // - ra nối tiếp

74194: 4 bit, vào // hay nối tiếp – ra // , nạp đồng bộ, dịch phải/trái.

SIPO (74ALS164)



PISO (74HC165)



(a)

Function Table

Inputs			Operation
SH/LD	CP	CP INH	
L	X	X	Parallel load
H	H	X	No change
H	X	H	No change
H	\bar{f}	L	Shifting
H	L	\bar{f}	Shifting

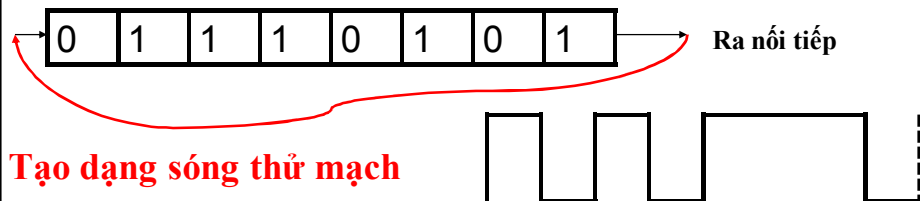
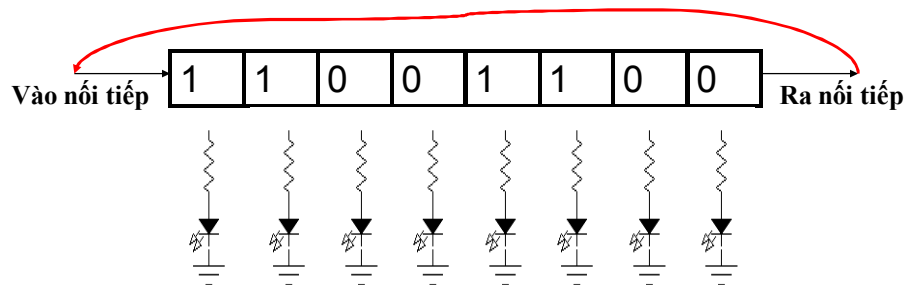
H = high level
L = low level
X = immaterial
 \bar{f} = PGT

(b)

IV. ỨNG DỤNG

- Lưu trữ và dịch chuyển dữ liệu.
- Tạo ký tự hay tạo các dạng điều khiển.
- Chuyển đổi dữ liệu từ nối tiếp ra song song hoặc ngược lại.

ĐIỀU KHIỂN DẠNG SÁNG TẮT CỦA ĐÈN



b_7	b_6	b_5	b_4	b_3	b_2	b_1	b_0	
1	0	1	1	0	1	0	0	$=180_{10}$

Dịch chuyển sang phải 1 bit

0	1	0	1	1	0	1	0	0	$=90_{10}$
---	---	---	---	---	---	---	---	---	------------

Dịch chuyển sang trái 1 bit

1	0	1	1	0	1	0	0	0	$=360_{10}$
---	---	---	---	---	---	---	---	---	-------------

Dịch chuyển sang trái, phải 1 bit là nhân đôi, chia đôi số thập phân tương ứng

Ứng dụng chuyển đổi song song sang nối tiếp và ngược lại để truyền dữ liệu giữa hai máy tính

